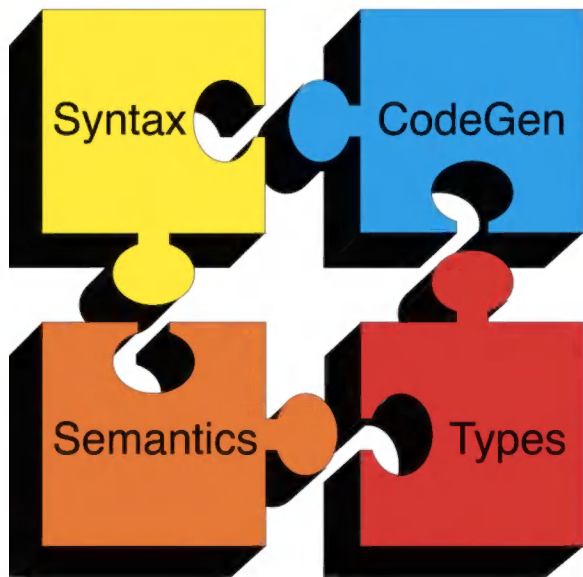




Compilers

Java Arrays

Assume $B < A$. What happens in the following?

```
B[] b = new B[10];
```

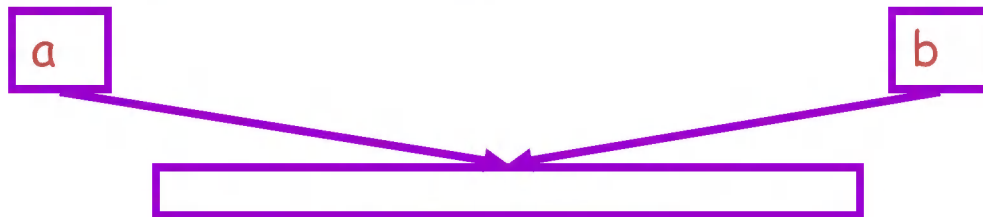
```
A[] a = b;
```

```
a[0] = new A();
```

```
b[0].aMethodNotDeclaredInA();
```

$B < A$ if B inherits from A as in Cool
 $C < A$ if $C < B$ and $B < A$ as in Cool
 $B[] < A[]$ if $B < A$ not as in Cool

```
B[] b = new B[10];  
A[] a = b;  
a[0] = new A();  
b[0].aMethodNotDeclaredInA();
```



Having multiple aliases to updateable locations with different types is unsound!

- Standard solution
 - Disallow subtyping through arrays

$B < A$ if B inherits from A

$C < A$ if $C < B$ and $B < A$

$B[] < A[]$ if $B = A$

- Java fixes the problem by checking each array assignment at runtime for type correctness
 - Is the type of the object being assigned compatible with the type of the array?
- Adds overhead on array computations
- But note: arrays of primitive types unaffected
 - Primitive types are not classes